

FACE RECOGNITION (PENGENALAN WAJAH) UNTUK SISTEM PRESENSI BERBASIS WEBSITE PADA LABORATORIUM SOFTWARE UNIVERSITAS ANDI DJEMMA

Musyawirul Haq^{*}, Dasril², Muhlis Muhallim³

^{1,2}Teknik Informatika/Universitas Andi Djemma; Jl. Tandipau, Kota Palopo;

Keywords:

Face Recognition, FaceNet, DeepFace, Implementasi, R&D, Blackbox.

Correspondent Email:

syawirhaq.hasbi@gmail.com

Abstrak. Penelitian ini dibuat sebagai upaya untuk mensupport peningkatan efisiensi dan juga modernisasi melakukan presensi di Laboratorium Software Fakultas Teknik Universitas Andi Djemma. Sejalan dengan berkembangnya teknologi informasi, maka dibutuhkan suatu sistem presensi yang dapat membantu mempermudah pencatatan kehadiran yang presisi serta aman. Oleh karena itu, tujuan penelitian ini untuk membuat rancangan bangun merealisasikan sistem presensi berbasis web yang menggunakan teknologi pengecekan wajah atau yang biasa disebut face recognition. Metode penelitian yang dilakukan yaitu *Research and Development (R&D)* dengan model pengembangan *Waterfall* yang meliputi tahap analisis, perancangan, kode, uji, dan implementasi system. Sistem ini dikembangkan menggunakan HTML, PHP, dan *Javascript* dengan pustaka *face-api.js* berbasis *TensorFlowJS* dan model *deep learning* pra-terlatih *FaceNet* dan *DeepFace*. Pengujian sistem menggunakan metode *blackbox* untuk memastikan seluruh fungsi sistem berjalan sesuai kebutuhan pengguna. Berdasarkan hasil penelitian, sistem presensi yang berkembang mendeteksi dan mengenali wajah secara *real-time* memiliki tingkat akurasi yang baik dan mendukung pengelolaan data presensi terintegrasi. Diharapkan dengan adanya sistem ini dapat bermanfaat dalam hal meningkatkan efektivitas pengelolaan presensi dan mendukung transformasi digital lingkungan laboratorium.

Abstract. *This research was conducted as an effort to support efficiency improvement and modernization of attendance recording in the Software Laboratory of the Faculty of Engineering, Andi Djemma University. In line with the development of information technology, an attendance system is needed to facilitate accurate and secure attendance recording. Therefore, the purpose of this research is to develop a web-based attendance system that uses face recognition technology. The research method used is Research and Development with a Waterfall development model, which includes the stages of analysis, design, coding, testing, and system implementation. This system was developed using HTML, PHP, and Javascript with the face-api.js library based on TensorFlowJS and pre-trained deep learning models FaceNet and DeepFace. The system was tested using the black box method to ensure that all system functions ran according to user requirements. Based on the research results, the developed attendance system detects and recognizes faces in real-time with a good level of accuracy and supports integrated attendance data management. It is hoped that this system will be useful in improving the effectiveness of attendance management and supporting the digital transformation of the laboratory environment.*



Copyright © [JITET](http://www.jitet.org) (Jurnal Informatika dan Teknik Elektro Terapan). This article is an open access article distributed under terms and conditions of the Creative Commons Attribution (CC BY NC)

1. PENDAHULUAN

Pesatnya perkembangan teknologi dalam era digital tersebut mempengaruhi juga sistem manajemen kehadiran dalam institusi pendidikan maupun perusahaan. Bahkan, meskipun metode konvensional yakni tanda tangan manual atau absensi lewat kartu masih kerap digunakan, ia mendatangkan berbagai kesulitan, baik dalam hal efisiensi waktu yang digunakan, ketidakamanan dari sisi kecurangan, dan risiko hilangnya atau rusaknya data dalam jumlah besar apabila tidak tertata atau diatur sesuai dengan aturan yang berlaku.[1]

Untuk alasan ini, keperluan untuk memiliki sistem presensi yang canggih, yang mampu beroperasi sejalan dengan kecepatan kondisi operasional apa pun dan memberikan pengukur akurat tentang keberadaan dan ketidakhadiran, terus meningkat. Pengembangan aplikasi presensi sepenuhnya menjadi kebutuhan daripada pilihan, sebagaimana dibuktikan oleh sejarah dan penelitian yang terbukti saling berkaitan. Pandemi global membuktikan bahwa dengan teknologi non-kontak adalah cara yang lebih baik dan lebih aman bagi negara untuk memperdagangkan data, hal ini memperkuat kebutuhan akan teknologi pengenalan wajah. Saatnya menggunakan aplikasi ini untuk otomatisasi manajemen kehadiran yang membantu meningkatkan sistem, data, keamanan, meminimalkan kelakuan penggunas sistem.[2]

Sebagai tanggapan terhadap sejumlah permasalahan dari sistem presensi konvensional, teknologi pengenalan wajah telah menjadi pilihan inovatif dan berpotensi. Dengan menggunakan algoritma kecerdasan buatan, teknologi ini didesain untuk mengenali wajah secara akurat. Secara relatif terhadap bentuk biometrik berbasis lainnya seperti biometrik jejak tangan atau RFID, sistem ini lebih baik karena tidak melibatkan kontak fisik dan pemrosesan data berlangsung cepat dan efisien.[3]

Selain uraian di atas, salah satu alasan kuat peneliti mengangkat judul penelitian ini adalah urgensi yang terjadi di Laboratorium *Software*, Program Studi Teknik Informatika, Fakultas Teknik, Universitas Andi Djemma.

Berdasarkan observasi dan pengalaman praktikum, peneliti menemukan bahwa sistem presensi masih dilakukan secara konvensional, sehingga belum ada penerapan sistem presensi berbasis teknologi informasi yang secara signifikan mendukung kemajuan dibidang tersebut. Hal ini mengindikasikan perlunya inovasi segera.

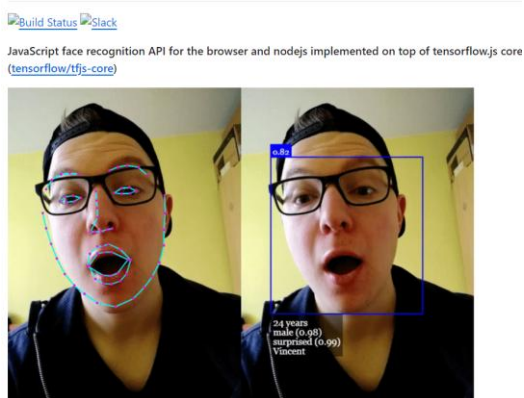
Dengan implementasi teknologi ini, diharapkan dapat tercipta sebuah solusi yang jauh lebih efisien untuk kebutuhan presensi, sekaligus memberikan kontribusi nyata dalam mendukung transformasi digital di bidang manajemen kehadiran pada institusi pendidikan maupun organisasi lainnya.

2. TINJAUAN PUSTAKA

2.1. *Pustaka face-api.js*

Pada proyek yang akan dibuat menggunakan *JavaScript* disini tugasnya sebagai jembatan untuk penggunaan *library* yang bernama *face-api.js* seperti yang ditunjukkan pada Gambar 1, ini merupakan API (*Application Programming Interface*) pengenalan wajah *Javascript* bersumber terbuka menggunakan *TensorFlowJS*. API ini menyediakan beberapa fungsi seperti deteksi wajah, analisis ekspresi wajah, prediksi usia dan jenis kelamin, deteksi tengara wajah, dan pengenalan wajah. Terdapat total 18 *file* dalam direktori "*weights*", seperti yang ditunjukkan pada Gambar 2, dalam direktori "*weights*" pada API, kombinasi deteksi wajah, deteksi landmark wajah, dan model ekstraksi fitur diperlukan untuk mengaktifkan pengenalan wajah. Untuk tugas deteksi wajah, *file* yang diperlukan adalah "*ssd_mobilenetv1_model-shard1*", "*ssd_mobilenetv1_model-shard2*" dan "*ssd_mobilenetv1_model-weights_manifest.json*". Untuk tugas deteksi landmark wajah, *file* yang diperlukan adalah "*face_landmark_68_tiny_model-shard1*" dan "*face_landmark_68_tiny_modelweights_manifest.json*". Untuk tugas ekstraksi fitur, *file* yang diperlukan adalah "*face_recognition_model-shard1*", "*face_recognition_model shard2*"-dan-"*face_recognition modelweight manifest.json*," Tabel

1 menunjukkan deskripsi singkat tentang tiga model pra-pelatihan *deep learning* yang diperlukan. Model-model tersebut didasarkan pada model-model yang sudah ada dan diintegrasikan ke dalam sistem yang diusulkan.[3] `face-api.js`



Gambar 1 `face-api.js`



Gambar 2 Bobot & Metadata Model Pra-pelatihan *deep learning* dalam direktori "weight" API

No.	Tugas	Algoritma <i>Deep Learning</i> dari <i>Pre-trained</i> Model	Tujuan
1.	Pendeteksian Wajah	SSD MobileNet V1	Mendeteksi lokasi wajah dan mengembalikan skor, koordinat, lebar, dan tinggi area wajah
2.	Landmark pendeteksian wajah	Depth-wise konvolusi yang dapat dipisahkan dengan blok padat yang terhubung	Mendeteksi 68 titik <i>Landmark</i> dan mengembalikan koordinat yang akan direferensikan untuk pra-pemrosesan penyetaraan wajah
3.	Ekstraksi Fitur	ResNet-34 seperti arsitektur	Mengekstrak 128 nilai fitur yang dinormalisasi vektor dari ROI wajah

Tabel 1 Deskripsi Model *Deep Learning*

2.2. Pengujian Black Box

Rahadi, Nur Wahyu, dan Cahya Vikasari [4] mendefinisikan *blackbox testing* sebagai teknik pengujian perangkat lunak yang difokuskan pada pengujian aspek fungsional sistem. Teknik ini digunakan untuk menemukan berbagai permasalahan, seperti fungsi yang tidak berjalan atau tidak tersedia, kesalahan

pada struktur data, akses basis data yang tidak tepat, kesalahan antarmuka, penurunan kinerja, serta kesalahan pada proses inialisasi dan terminasi sistem.

Sementara itu, Arwaz dkk.[5] menyebutkan bahwa pengujian *blackbox* dilakukan tanpa memperhatikan proses internal atau kode sumber perangkat lunak, melainkan berdasarkan hubungan antara input yang diberikan dan output yang dihasilkan.

Dari kedua pendapat tersebut, dapat disimpulkan bahwa *blackbox testing* merupakan metode pengujian yang bertujuan untuk memastikan sistem yang dikembangkan telah berjalan sesuai kebutuhan dan tidak mengandung kesalahan pada tahap implementasi maupun pengujian.

2.3. Unified Modeling Language

Menurut Rosa dan Shalaluddin [6], UML merupakan bahasa gambar yang menggambarkan sistem dan mengekspresikan informasi kualitatif melalui berbagai jenis teks dan diagram. UML adalah industri yang diterima dan disepakati secara luas untuk menganalisa, merancang, dan mendokumentasikan perangkat lunak. UML berfungsi untuk menggambarkan proses aplikasi. Diagram UML yang jelas ditampilkan di bawah:

a. Usecase Diagram

Kasus penggunaan dapat didefinisikan dari perspektif pengguna sebagai apa yang akan dilakukan oleh sistem dan bagaimana pengguna berinteraksi dengannya untuk membuatnya bekerja. Dengan cara itu, use case adalah serangkaian kelompok yang saling terkait yang terorganisir sistem yang berjalan atau dipantau oleh para peserta. Simbol usecase ditunjukkan pada tabel 2:

No	Simbol	Nama	Keterangan
1		Actor	Menspesifikasikan sekumpulan peran yang dimainkan oleh pengguna saat berinteraksi dengan use case.
2		Dependency	Hubungan di mana perubahan pada elemen independen akan berdampak pada elemen yang tergantung padanya elemen yang tidak mandiri.
3		Generalization	Hubungan di mana objek anak (descendensi) mewarisi struktur data dengan objek di atasnya, yaitu objek induk (ancestor).
4		Include	Menspesifikasikan bahwa use case yang menjadi sumber dinyatakan secara eksplisit.
5		Extend	Menspesifikasikan bahwa use case target memperluas perilaku dari use case sumber di titik tertentu.
6		Association	Mengaitkan satu objek dengan objek yang lain.
7		Sistem	Menspesifikasikan paket yang menunjukkan sistem dengan cara yang terbatas.
8.		Use Case	Deskripsi dari rangkaian tindakan yang ditampilkan oleh sistem yang menghasilkan suatu hasil yang dapat diukur untuk seorang aktor.

Tabel 2 Usecase Diagram [6]

b. Activity Diagram

Banyaknya kasus menggunakan diagram aktivitas untuk menunjukkan logika lingkungan proses, proses bisnis, dan alur kerja. Yang paling penting, diagram aktivitas sepadan dengan operasi sistem pada perangkat lunak, tapi bukan suatu tindakan pengguna. Dalam beberapa kasus, sistem dapat menyelesaikan aktivitasnya. Gambar Aktivitas beberapa simbol yang digunakan dalam membuat diagram aktivitas pada Tabel 3 adalah sebagai berikut.

No	Simbol	Nama	Keterangan
1		Aktivitas	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
2		Percabangan	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
3		Status awal	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
4		Status akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki status akhir.
5		Swimlane	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

Tabel 3 Activity Diagram [6]

c. Sequence Diagram

Diagram sequence menggunakan berbagai objek contoh dan pesan yang diatur di antara mereka dalam skenario penggunaan untuk menjelaskan perilaku dalam suatu kondisi. Elemen utama diagram ini adalah objek yang ditulis dalam kotak persegi panjang yang mewakili pesan dan ditandai dengan garis panah dan perkembangan vertikal. tabel 4 menampilkan simbol-simbol dalam diagram sequence.

No	Simbol	Nama	Keterangan
1		LifeLine	Objek entitas antar muka yang saling berinteraksi.
2		Message	Detail dari komunikasi antara objek yang berisi data mengenai kegiatan yang berlangsung.
3		Message	Detail dari komunikasi antara objek yang berisi informasi-informasi tentang aktifitas yang sedang berlangsung.

Tabel 4 Sequence Diagram [6]

2.4. Penelitian Pengembangan

Penelitian dan pengembangan adalah metode ilmiah yang dirancang untuk menciptakan produk atau model inovatif guna menyelesaikan masalah dalam kehidupan manusia. Metode ini dimulai dengan analisis masalah sebelum proses perancangan, pengembangan, dan pengujian produk dilakukan berulang kali untuk memastikan tingkat efektivitas dan keefektifan yang tinggi. Penelitian dan pengembangan banyak

diterapkan dalam bidang pendidikan untuk menciptakan model kepemimpinan yang lebih baik, modul pelatihan, dan bahan pembelajaran. [7]

Fungsi R&D semakin dipandang sebagai penggerak utama bagi inovasi berkelanjutan dan daya saing jangka panjang. Sarpong dan rekan-rekan memperkenalkan kerangka “*three pointers*” yang menekankan pentingnya penyaluran antara investasi, kualitas sumber daya manusia, serta kapasitas organisasi untuk terus belajar. Ketiga elemen tersebut dianggap krusial dalam membangun sistem inovasi yang mampu mendukung transisi iklim dan menghasilkan inovasi yang benar-benar transformatif, bukan sekadar peningkatan bertahap. [8] R&D dipahami sebagai mekanisme pembentukan pengetahuan dan penguatan *absorptive capacity*, yang memungkinkan organisasi mengidentifikasi peluang teknologi baru serta menjalin kolaborasi dengan universitas dan mitra eksternal demi mendorong eco-innovation.[8] Pada tataran nasional, belanja R&D menjadi faktor strategis yang memengaruhi keberhasilan agenda pembangunan berkelanjutan dan posisi kompetitif suatu negara di tengah dinamika persaingan global. [9]

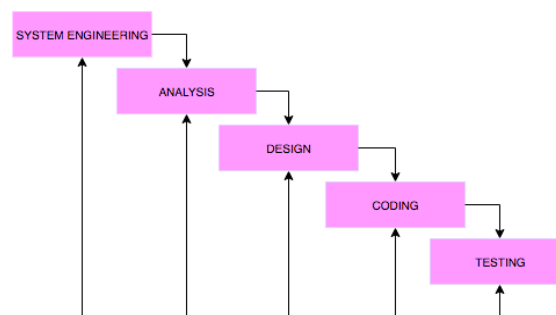
Berbagai temuan empiris mutakhir menunjukkan bahwa pengaruh aktivitas R&D terhadap kinerja perusahaan bersifat *trade-off* antarwaktu. Penelitian pada perusahaan di Pakistan dan Tiongkok mengindikasikan bahwa intensitas R&D cenderung menekan profitabilitas jangka pendek (ROA) karena tingginya biaya serta adanya jeda menuju komersialisasi hasil riset. Namun, dalam jangka panjang, investasi ini justru meningkatkan nilai pasar dan kinerja berbasis valuasi (*Tobin's Q, firm value*).[10] Pola serupa teridentifikasi pada perusahaan di Slovenia maupun perusahaan-perusahaan global yang berorientasi R&D pengeluaran untuk riset kurang memberikan hasil langsung, tetapi berkontribusi positif pada kinerja dalam periode waktu yang lebih panjang.[9] Studi lain bahkan mencatat bahwa peningkatan belanja R&D dapat menurunkan kinerja keuangan saat ini, meskipun dampak negatif tersebut cenderung lebih lemah pada BUMN dan di pasar dengan tingkat persaingan tinggi.[11] Secara keseluruhan, bukti ini menegaskan bahwa R&D merupakan investasi strategis jangka panjang yang memerlukan

kesiapan perusahaan menghadapi penurunan laba sementara.

2.5. Model Waterfall

Model *Waterfall* merupakan pendekatan klasik dalam *Software Development Life Cycle* (SDLC) yang mengikuti alur kerja linear dan berurutan. Seluruh tahap, mulai dari perencanaan, analisis, perancangan, implementasi, pengujian, hingga pemeliharaan yang diselesaikan secara bertahap, dengan keluaran tiap fase menjadi input bagi fase berikutnya.[12][13][14]. Suatu tahap baru dapat dimulai hanya setelah tahap sebelumnya dirampungkan, dan tidak tersedia mekanisme kembali ke fase sebelumnya tanpa prosedur formal tertentu.[12][13] Struktur yang berlapis dan teratur ini menjadikan *Waterfall* dikenal sebagai *linear sequential* model atau model siklus hidup tradisional, yang menekankan proses pengembangan perangkat lunak yang sistematis, terstruktur, serta terdokumentasi dengan baik.[14][15]

Fitria dkk.[16] menjelaskan bahwa model *Waterfall* merupakan pendekatan klasik yang bersifat sistematis dan bertahap dalam pengembangan perangkat lunak. Model ini awalnya dikenal sebagai *linear sequential* model dan digunakan untuk menggambarkan siklus hidup perangkat lunak secara tradisional, sebagaimana pertama kali diperkenalkan oleh Winston Royce pada tahun 1970-an. Meskipun dianggap sebagai model lama, pendekatan ini tetap banyak diterapkan dalam praktik pengembangan perangkat lunak. Prosesnya berlangsung secara runut, di mana setiap fase harus diselesaikan secara tuntas sebelum melanjutkan ke fase berikutnya. Karakteristik utama *Waterfall* adalah penekanan pada penyelesaian penuh tiap tahap sebelum memasuki tahapan selanjutnya. Ilustrasi mengenai model *Waterfall* ditampilkan pada Gambar 3



Gambar 3 Model *Waterfall* [16]

Berdasarkan uraian tersebut, dapat disimpulkan bahwa metode ini menuntut penyelesaian setiap fase secara runut, di mana suatu fase baru dapat dimulai hanya setelah fase sebelumnya diselesaikan sepenuhnya, tanpa fleksibilitas untuk kembali ke tahap yang telah dilewati. Penelitian ini mengadopsi model *Waterfall* karena penerapannya relatif sederhana dalam proses pengembangan sistem dan sesuai untuk perangkat lunak dengan kebutuhan yang telah terdefinisi dengan jelas sejak awal, sehingga dapat meminimalkan potensi kesalahan.

2.6. Implementasi

Implementasi merupakan fase penerjemahan rancangan atau konsep ke dalam tindakan operasional agar suatu sistem, program, atau kebijakan dapat berfungsi sesuai tujuan yang ditetapkan. Pada tahap ini, berbagai sumber daya mulai dari tenaga kerja, teknologi, anggaran, hingga prosedur diatur dan diselaraskan untuk merealisasikan rencana yang telah disusun. Proses implementasi tidak hanya menjalankan instruksi dalam dokumen perencanaan, tetapi juga melibatkan penyesuaian di lapangan, proses uji, pemantauan, serta evaluasi untuk memastikan hasil yang dicapai efektif dan efisien. Tingkat keberhasilannya sangat bergantung pada kesiapan pelaksana, kejelasan arahan, ketersediaan fasilitas pendukung, serta dukungan organisasi dan para pemangku kepentingan.[17]

3. METODE PENELITIAN

3.1. Jenis Data dan Sumber Data

3.1.1. Jenis Data

Jenis data kualitatif pada penelitian ini didapatkan dari hasil wawancara dengan narasumber. Sedangkan jenis data kuantitatif pada penelitian ini didapatkan melalui kuesioner yang diisi oleh narasumber.

3.1.2. Sumber Data

Data primer pada penelitian ini didapatkan dari hasil observasi, wawancara, dan hasil pengisian kuesioner. Sedangkan data sekunder pada penelitian ini berasal dari studi literatur, penelitian yang relevan, internet, dan sumber-sumber lainnya yang berkaitan dengan topik penelitian.

3.2. Tempat dan Waktu Penelitian

Tempat penelitian ini dilakukan di Laboratorium *Software*, Program Studi Teknik Informatika, Fakultas Teknik, Universitas Andi Djemma, Jl. Tandipau, Tomarundung, Kec. Wara Barat, Kota Palopo, Sulawesi Selatan 91913.

Waktu penelitian dilakukan pada bulan Februari sesuai dengan rincian pada tabel 3.

No	Tahapan Penelitian	Februari				Maret				April			
		1	2	3	4	1	2	3	4	1	2	3	4
1	Analisis	■	■	■	■								
2	Desain			■	■	■	■	■	■				
3	Coding					■	■	■	■	■	■	■	■
4	Testing									■	■	■	■
5	Implementasi											■	■

Tabel 3 Waktu Penelitian

3.3. Prosedur Penelitian

Penelitian ini menggunakan prosedur penelitian dengan model pengembangan *Waterfall*.

Pada uraian berikut dijelaskan prosedur penelitian sesuai dengan poin-poin yang ada model pengembangan *Waterfall* :

a. System Engineering

Tahap pertama yang dilakukan peneliti adalah melakukan pengumpulan data di lokasi penelitian. Pada tahapan ini juga peneliti menentukan kebutuhan yang diperlukan dalam pengembangan sistem pada penelitian ini

b. Analysis

Setelah data sudah didapatkan dari tahapan pertama, peneliti kemudian menganalisa data-data tersebut untuk dijadikan patokan dalam pengembangan aplikasi.

c. Design

Setelah tahap kedua atau analisis selesai dilakukan dan sudah menghasilkan data berupa hasil analisis data dan kebutuhan sistem, peneliti kemudian membuat rancangan awal sebagai referensi awal yang digunakan dalam pengembangan sistem seperti *user interface* dan diagram UML dari sistem yang akan dibuat.

d. Coding

Pada tahap ini peneliti sudah mengembangkan sistem dengan berbekal data dan rancangan atau desain awal dari tahapan sebelumnya dengan melakukan pengkodean atau *coding*.

e. Testing

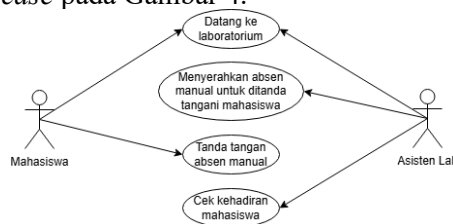
Setelah tahapan *coding* atau pengkodean selesai dilaksanakan, tahapan selanjutnya yaitu tahap *testing*, pada tahap ini peneliti melakukan pengetesan terhadap sistem yang telah dikembangkan berbekal data dari setiap tahapan sebelumnya serta memastikan apakah sudah berjalan sesuai dengan rancangan awal atau belum, jika masih terdapat *error* ataupun *bug* maka peneliti melakukan perbaikan sampai tidak ada *error* dan *bug* lagi.

f. Implementation (implementasi)

Jika sudah sampai pada tahap implementasi, artinya sistem telah selesai dikembangkan dan sistem juga sudah diimplementasikan atau digunakan pada lokasi penelitian.

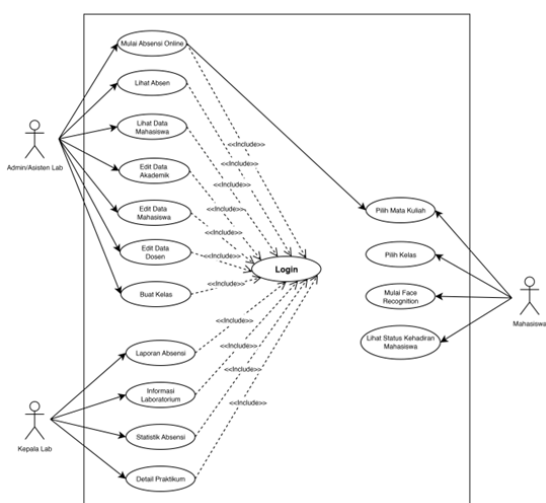
3.4. Analisis Sistem yang Berjalan

Sistem yang berjalan pada Lokasi penelitian saat ini dapat dilihat pada diagram *usecase* pada Gambar 4.



Gambar 4 Usecase Sistem yang Berjalan

Gambar diatas merupakan diagram *use case* dari sistem yang berjalan saat ini pada lokasi penelitian dimana proses absensi kehadiran masih menggunakan metode manual.



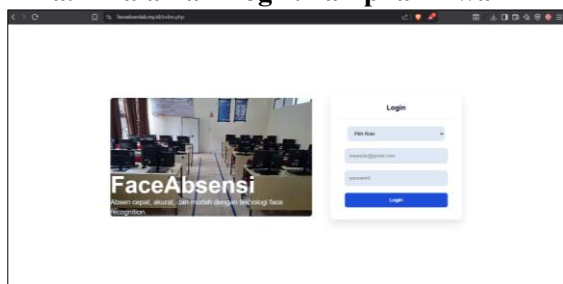
Gambar 7 Usecase diagram Sistem

Pada Gambar 7 terdapat 3 aktor dalam sistem yaitu Admin atau Asisten Lab, Kepala Laboratorium dan Mahasiswa. Aktor Admin atau Asisten Lab setelah login kedalam sistem, memiliki akses seperti memulai absensi, melihat absen, melihat dan mengedit data mahasiswa, edit data akademik, edit data dosen dan membuat kelas, untuk aktor Kepala Lab setelah login ke dalam sistem kemudian dapat melihat laporan absensi serta dapat melakukan export laporan absensi dalam bentuk file .xlsx. Sementara aktor mahasiswa sebelum memulai absensi, terlebih dahulu memilih mata kuliah setelah itu memilih kelas dan kemudian mulai melakukan Absensi online dengan face recognition. Setelah melakukan absensi, mahasiswa dapat melihat status kehadiran mahasiswa.

4.1.3. Coding

Berikut merupakan tampilan Website Face Absensi hasil dari pembuatan program yang telah dilakukan.

a. Halaman Login/Tampilan Awal

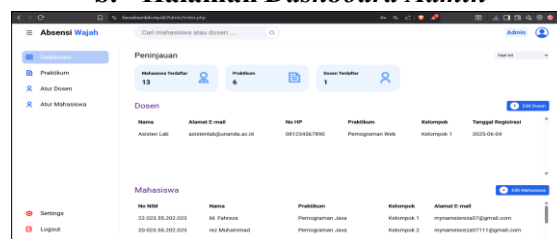


Gambar 8 Halaman Login

Pada Gambar 8 merupakan tampilan halaman Login yang pertama kali muncul saat user mengakses link website. User memasukkan

username dan password sesuai role untuk masuk ke dalam website.

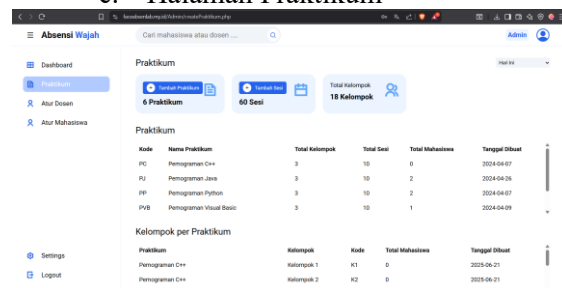
b. Halaman Dashboard Admin



Gambar 9 Halaman Dashboard Admin

Pada Gambar 9 Merupakan tampilan halaman Dashboard admin yang muncul ketika admin telah login menggunakan username dan password dengan role admin. Pada halaman ini admin dapat melihat beberapa menu dan informasi seperti mahasiswa, matakuliah dan dosen.

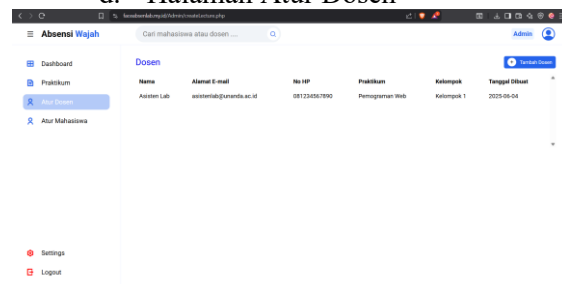
c. Halaman Praktikum



Gambar 10 Halaman Praktikum

Pada Gambar 10 merupakan tampilan halaman praktikum yang akan muncul ketika admin mengakses menu praktikum. Pada halaman ini, admin bisa membuat jadwal praktikum baru.

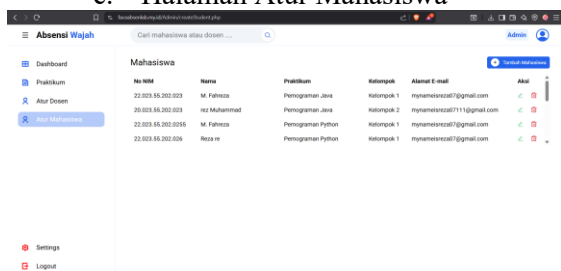
d. Halaman Atur Dosen



Gambar 11 Halaman Atur Dosen

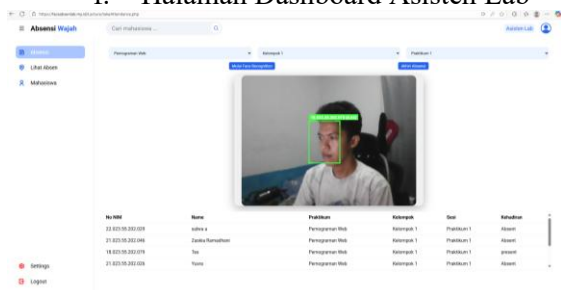
Pada Gambar 11 merupakan tampilan halaman atur dosen, halaman ini akan muncul ketika admin memilih menu atur dosen pada sidebar menu. Pada halaman ini admin bisa mengatur data dosen dengan praktikum dan kelompok masing-masing.

e. Halaman Atur Mahasiswa



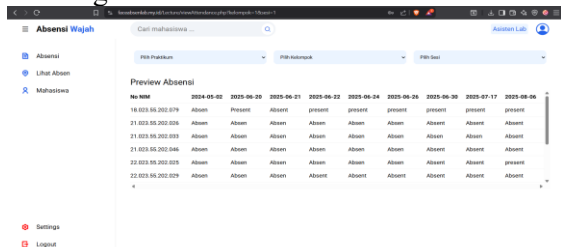
Gambar 12 Halaman Atur Mahasiswa
 Halaman Atur Mahasiswa pada Gambar 12 merupakan tampilan halaman tempat admin mengatur data mahasiswa atau praktikan yang akan melaksanakan praktikum pada laboratorium software berdasarkan Lab atau praktikum masing-masing mahasiswa.

f. Halaman Dashboard Asisten Lab



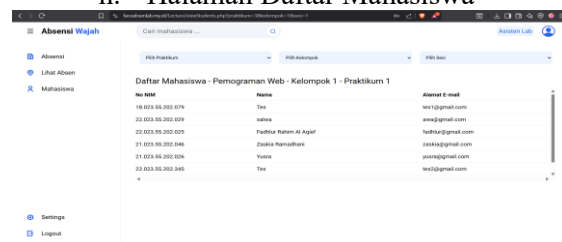
Gambar 13 Halaman Dashboard Asisten Lab
 Gambar 13 merupakan tampilan halaman dashboard untuk asisten laboratorium ketika login menggunakan *username*, *password* dan *role* asisten laboratorium. Pada halaman ini, asisten laboratorium dapat memulai absensi *face recognition* dengan terlebih dahulu memilih praktikum, kelompok serta sesi yang akan dimulai, setelah itu asisten dapat memulai absensi dan mengakhiri setelah sesi selesai.

g. Halaman Lihat Absen



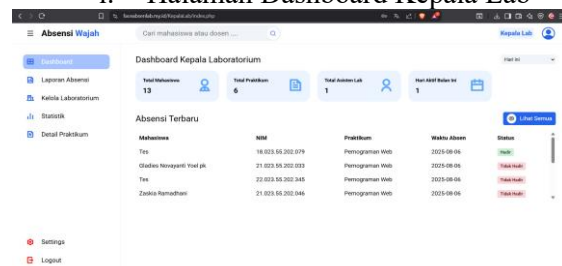
Gambar 14 Halaman Lihat Absen
 Gambar 14 adalah tampilan halaman Lihat absen, pada halaman ini asisten lab dan mahasiswa dapat melihat *preview* absensi yang telah dilakukan sesuai dengan masing-masing praktikum, kelompok dan sesi.

h. Halaman Daftar Mahasiswa



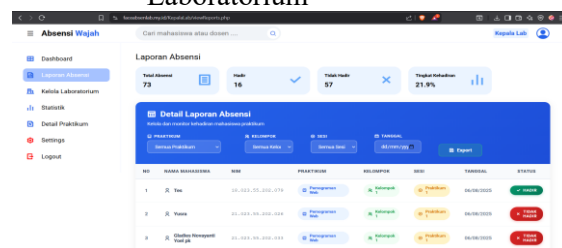
Gambar 15 Halaman Daftar Mahasiswa
 Halaman Daftar Mahasiswa pada Gambar 15 merupakan halaman dimana asisten lab dapat melihat daftar mahasiswa berdasarkan praktikum, kelompok dan sesi masing-masing.

i. Halaman Dashboard Kepala Lab



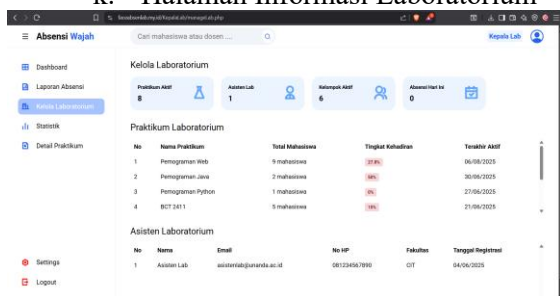
Gambar 16 Halaman Dashboard Kepala Lab
 Halaman Dashboard kepala laboratorium pada Gambar 16 merupakan halaman yang berisi informasi terkait absensi mahasiswa terbaru, total mahasiswa, total praktikum, total asisten laboratorium, dan hari aktif bulan ini.

j. Halaman Laporan Absensi Kepala Laboratorium



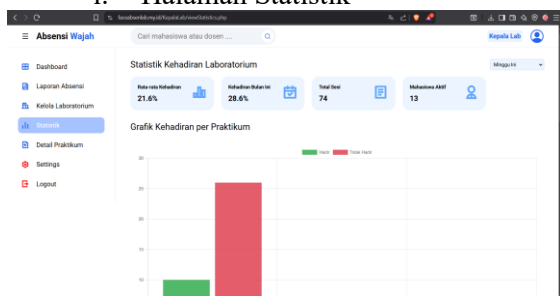
Gambar 17 Halaman Absensi Kepala Lab
 Halaman laporan absensi pada Gambar 17 berisi laporan absensi seperti total absensi, total mahasiswa hadir, total mahasiswa tidak hadir, dan persentase tingkat kehadiran. Selain itu terdapat detail laporan absensi mahasiswa berdasarkan praktikum, kelompok, sesi, dan tanggal praktikum, pada halaman ini jg terdapat fitur *export* hasil absensi kedalam bentuk file dengan ekstensi *.xlsx*.

k. Halaman Informasi Laboratorium



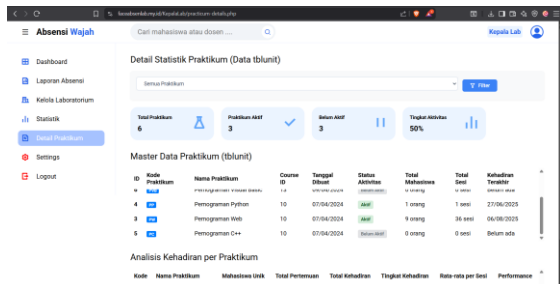
Gambar 18 Halaman Informasi Laboratorium Halaman informasi laboratorium pada Gambar 18 adalah halaman yang berisi tentang informasi apa semua yang ada pada laboratorium, seperti jumlah praktikum aktif, jumlah asisten laboratorium, jumlah kelompok aktif, dan absensi hari ini. Selain itu terdapat juga informasi tentang nama praktikum apa saja yang ada di laboratorium beserta jumlah peserta, persentase tingkat kehadiran, dan terakhir aktif bagi masing-masing praktikum tersebut.

l. Halaman Statistik



Gambar 19 Halaman Statistik Halaman statistik pada Gambar 19 berisi informasi tentang statistik kehadiran di laboratorium software. Pada halaman ini berisi persentase rata-rata kehadiran, persentase kehadiran bulan ini, total sesi dan total mahasiswa aktif. Selain itu terdapat juga grafik kehadiran harian, statistik detail per praktikum, dan praktikum belum aktif.

m. Halaman Detail Praktikum



Gambar 20 Halaman Detail Praktikum

Halaman detail praktikum pada Gambar 20 merupakan halaman yang dapat diakses oleh kepala laboratorium yang berisi informasi tentang detail praktikum seperti total praktikum, praktikum aktif, praktikum belum aktif, dan persentase tingkat aktivitas. Selain itu terdapat pula master data praktikum dan analisis kehadiran per praktikum.

4.1.4. Testing (Pengujian)

Pada tahap ini peneliti melakukan pengujian terhadap sistem yang telah dibuat dan dikembangkan pada tahap-tahap sebelumnya. Berdasarkan penjelasan peneliti pada BAB 3, pada tahap pengujian ini peneliti menggunakan Black box testing dengan teknik all pair testing atau pairwise testing. Berikut merupakan uraian dari hasil pengujian yang telah peneliti lakukan:

Page	Testing Function	Status
Login	Pilih Role	(✓) Succeed
	Input Email	(✓) Succeed
	Input Password	(✓) Succeed
	Login Button	(✓) Succeed
Dashboard Admin	Cari Mahasiswa atau Dosen	(✓) Succeed
	Edit Dosen	(✓) Succeed
	Edit Mahasiswa	(✓) Succeed
	Sidebar Dashboard Button	(✓) Succeed
	Sidebar Praktikum Button	(✓) Succeed
	Sidebar Atur Dosen Button	(✓) Succeed
	Sidebar Atur Mahasiswa Button	(✓) Succeed
Praktikum Admin	Sidebar Logout Button	(✓) Succeed
	Short date	(✓) Succeed
	Tambah Praktikum Button	(✓) Succeed
Atur Dosen Admin	Tambah Sesi Button	(✓) Succeed
	Tambah Dosen Button	(✓) Succeed
Atur Mahasiswa Admin	Tambah Mahasiswa Button	(✓) Succeed
	Delete Button	(✓) Succeed
Dashboard Asisten Lab	Edit Button	(✓) Succeed
	Pilih Kelompok	(✓) Succeed
	Pilih Sesi	(✓) Succeed
	Mulai Face Recognition Button	(✓) Succeed
	Open Camera	(✓) Succeed
	Akhiri Absensi Button	(✓) Succeed
	Sidebar Dashboard Button	(✓) Succeed
	Sidebar Praktikum Button	(✓) Succeed
	Sidebar Atur Dosen Button	(✓) Succeed
	Sidebar Atur Mahasiswa Button	(✓) Succeed
Lihat Absen Asisten Lab	Sidebar Logout Button	(✓) Succeed
	Pilih Praktikum	(✓) Succeed
Mahasiswa	Pilih Kelompok	(✓) Succeed
	Pilih Sesi	(✓) Succeed
	Preview Absensi	(✓) Succeed
	Preview Data Mahasiswa	(✓) Succeed
Dashboard Kepala Lab	Preview Data Mahasiswa	(✓) Succeed
	Open page	(✓) Succeed
Laporan absensi kepala lab	Open page	(✓) Succeed
	Pilih praktikum	(✓) Succeed
	Pilih kelompok	(✓) Succeed
	Pilih sesi	(✓) Succeed
	Pilih tanggal	(✓) Succeed
	Export .xlsx	(✓) Succeed
	Show detail absensi	(✓) Succeed
	Sidebar Logout Button	(✓) Succeed
Kelola laboratorium	Open page	(✓) Succeed
Statistik	Open page	(✓) Succeed
Detail praktikum	Open page	(✓) Succeed
	Pilih praktikum	(✓) Succeed

Tabel 3 Pengujian Sistem

Selain menggunakan metode *Black Box Testing*, pada tahapan ini peneliti juga memberikan kuesioner kepada 10 responden terpilih dengan topik pertanyaan yang berkaitan dengan fungsionalitas sistem (hasil kuesioner terlampir). Setiap bulir pertanyaan pada kuesioner memiliki bobot jawaban dari 1,2,3 dan 4, untuk setiap jawaban yang dipilih oleh responden, berikut adalah bobot yang

didapatkan dari hasil kuesioner yang telah disebar:

Responden	Indikator/Variabel										Total Skor	Skor Maksimal
	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10		
R1	4	4	4	4	4	4	4	4	4	4	40	40
R2	4	4	4	4	4	4	4	4	4	4	40	40
R3	4	4	4	4	4	4	4	4	4	4	40	40
R4	4	4	4	4	4	4	4	4	4	4	40	40
R5	4	4	4	3	4	3	3	4	3	4	36	40
R6	4	4	4	3	3	4	4	4	4	4	38	40
R7	4	4	4	3	4	3	3	4	3	4	36	40
R8	4	3	3	4	3	2	4	3	3	4	33	40
R9	4	3	3	4	3	2	4	3	3	4	33	40
R10	3	3	3	3	2	3	3	3	3	3	29	40
Total											365	400

Tabel 4 Hasil Perhitungan Bobot Kuesioner Bobot yang telah didapatkan kemudian dihitung menggunakan rumus seperti di bawah:

$$\text{Persentase kelayakan} = \frac{365}{400} \times 100 = 91,25\%$$

Setelah mendapatkan hasil persentase dari bobot yang dihitung menggunakan rumus, kemudian nilai persentase tersebut dikonversi menggunakan interval rentang persentase berikut:

No.	Persentase	Keterangan
1	0% - 25%	Sangat Tidak Layak
2	26% - 50%	Tidak Layak
3	51% - 75%	Layak
4	76% - 100%	Sangat Layak

Tabel 5 Interval Rentang Persentase Hasil yang didapatkan adalah **91,25%** dimana nilai tersebut terdapat pada rentan persentase antara 76% - 100% dimana pada rentan tersebut keterangan yang didapatkan adalah Sangat Layak.

4.1.5. Pembahasan

Setelah semua tahap dari analisis, desain, coding, dan testing telah selesai dilaksanakan dan sistem absesnsi *face recognition* telah selesai dilakukan serta tidak ditemukan lagi *error* atau *bug* major pada sistem yang telah dikembangkan, maka *Face Recognition* Untuk Sistem Presensi Berbasis Web telah sampai pada tahap implementation atau penerapan pada lokasi penelitian dalam hal ini Laboratorium Software, Fakultas Teknik, Universitas Andi Djemma yang dilakukan dengan melibatkan laboran dan asisten Laboratorium Software

serta beberapa mahasiswa Program Studi Teknik Informatika. Hasil ini diperkuat oleh uji kelayakan yang telah dilakukan dengan menggunakan kuesioner terkait Rancang Bangun *Face Recognition* untuk Sistem Presensi Berbasis Web, yang menunjukkan hasil **91,25%**, dengan keterangan Sangat Layak berdasarkan *table* rentang interval presentase. Sehingga Sistem *Face Recognition* untuk sistem Presensi Berbasis Web sudah dapat diterapkan.

5. KESIMPULAN

Berdasarkan hasil tahapan penelitian yang telah dilakukan dalam pengembangan sistem *Face Recognition* Untuk Sistem Presensi Berbasis Web pada Laboratorium Software Fakultas Tnik Universitas Andi Djemma, maka peneliti menarik beberapa poin kesimpulan sebagai berikut:

- Pembuatan dan pengembangan sistem *Face Recognition* Untuk Sistem Presensi Berbasis Web telah selesai dilakukan. Hal ini dibuktikan dengan sudah dilakukan implementasi pada lokasi tersebut.
- Sistem *Face Recognition* untuk Sistem Presensi Berbasis Web sudah berjalan sesuai dengan rancangan dan rencana awal, sudah minim *bug* dan *error* dibuktikan dengan hasil pengujian menggunakan *black box testing* dengan teknik *all pair testing* serta hasil kuesioner yang telah disebar denan perolehan persentase sebesar 91,25%, dengan keterangan Sangat Layak.
- Sistem *Face Recognition* untuk Sistem Presensi Berbasis Web sudah dapat diterapkan di lokasi penelitian dan digunakan sebagai media absensi berbasis teknologi menggantikan sistem manual sebelumnya yang tentunya berjalan lurus dengan visi keilmuan program studi Teknik Informatika yang tentunya berbasis teknologi.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada pihak-pihak terkait yang telah memberi dukungan terhadap penelitian ini.

DAFTAR PUSTAKA

- [1] G. C. Rorimpandey, S. Agnes Intan, and Q. C. Kainde, "PENERAPAN ALGORITMA MULTI-TASK CASCADED CONVOLUTIONAL NEURAL NETWORK PADA SISTEM PRESENSI PENGAWAS SEKOLAH DI BIDANG COMPUTER VISION," *Jurnal Teknik Informatika (JUTIF)*, vol. 5, no. 4, pp. 593–600, 2024, doi: 10.52436/1.jutif.2024.5.4.2218.
- [2] C. Gomes, S. Chanchal, T. Desai, and D. Jadhav, "Class Attendance Management System using Facial Recognition," *ITM Web of Conferences*, vol. 32, p. 02001, 2020, doi: 10.1051/itmconf/20203202001.
- [3] N. A. Ismail *et al.*, "Web-based University Classroom Attendance System Based on Deep Learning Face Recognition," *KSI Transactions on Internet and Information Systems*, vol. 16, no. 2, pp. 503–523, Feb. 2022, doi: 10.3837/tiis.2022.02.008.
- [4] N. W. Rahadi and C. Vikasari, "Penguujian Software Aplikasi Perawatan Barang Milik Negara Menggunakan Metode Black Box Testing Equivalence Partitions," *Infotekmesin*, vol. 11, no. 1, pp. 57–61, Jan. 2020, doi: 10.35970/infotekmesin.v11i1.124.
- [5] A. Arifta Arwaz, K. Putra, R. Putra, T. Kusumawijaya, and A. Saifudin, "Jurnal Teknologi Sistem Informasi dan Aplikasi Pengujian Black Box pada Aplikasi Sistem Seleksi Pemenang Tender Menggunakan Teknik Equivalence Partitions," vol. 2, no. 4, pp. 2654–4229, 2019, [Online]. Available: <http://openjournal.unpam.ac.id/index.php/JT-SI/index130>
- [6] A. S. & M. S. Rosa, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek Edisi Revisi*. Informatika Bandung, 2019.
- [7] M. Waruwu, "Metode Penelitian dan Pengembangan (R&D): Konsep, Jenis, Tahapan dan Kelebihan," *Jurnal Ilmiah Profesi Pendidikan*, vol. 9, no. 2, pp. 1220–1230, May 2024, doi: 10.29303/jipp.v9i2.2141.
- [8] D. Sarpong, D. Boakye, G. Ofosu, and D. Botchie, "The three pointers of research and development (R&D) for growth-boosting sustainable innovation system," *Technovation*, vol. 122, p. 102581, 2023, doi: <https://doi.org/10.1016/j.technovation.2022.102581>.
- [9] D. Ravšelj and A. Aristovnik, "The impact of R and D expenditures on corporate performance: Evidence from Slovenian and world R and D companies," *Sustainability (Switzerland)*, vol. 12, no. 5, Mar. 2020, doi: 10.3390/su12051943.
- [10] T. Y. Leung and P. Sharma, "Differences in the impact of R&D intensity and R&D internationalization on firm performance – Mediating role of innovation performance," *J Bus Res*, vol. 131, pp. 81–91, Jul. 2021, doi: 10.1016/J.JBUSRES.2021.03.060.
- [11] Muhammad Arif Khan *et al.*, "Impact of R&D on Firm Performance: Do Ownership Structure and Product Market Competition Matter?," *Sage Open*, vol. 13, no. 4, p. 21582440231199560, Oct. 2023, doi: 10.1177/21582440231199560.
- [12] A. Saravanos and M. Curinga, "Simulating the Software Development Lifecycle: The Waterfall Model," *ArXiv*, vol. abs/2308.03940, p., 2023, doi: 10.48550/arxiv.2308.03940.
- [13] Y. Nugraha, "Information System Development With Comparison of Waterfall and Prototyping Models," *RISTEC: Research in Information Systems and Technology*, p., 2020, doi: 10.31980/ristec.v1i2.1202.
- [14] A. Z. D. N. Adiya, D. L. Anggraeni, and I. Albana, "Analisa Perbandingan Penggunaan Metodologi Pengembangan Perangkat Lunak (Waterfall, Prototype, Iterative, Spiral, Rapid Application Development (RAD))," *Merkurius: Jurnal Riset Sistem Informasi dan Teknik Informatika*, p., 2024, doi: 10.61132/merkurius.v2i4.148.
- [15] A. Agarwal, A. Agarwal, D. K. Verma, D. Tiwari, and R. Pandey, "A Review on Software Development Life Cycle," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, p., 2023, doi: 10.32628/cseit2390387.
- [16] O. Fitria, N. Hasanah, M. Pd, and R. S. Untari, *BUKU AJAR REKAYASA PERANGKAT LUNAK Diterbitkan oleh UMSIDA PRESS UNIVERSITAS MUHAMMADIYAH SIDOARJO 2020*. 2020.
- [17] I. N. Rizki, D. Prayoga, M. L. Puspita, and M. Q. Huda, "IMPLEMENTASI EXPLORATORY DATA ANALYSIS UNTUK ANALISIS DAN VISUALISASI DATA PENDERITA STROKE KALIMANTAN SELATAN MENGGUNAKAN PLATFORM TABLEAU," *Jurnal Informatika dan Teknik*

Elektro Terapan, vol. 12, no. 1, Jan. 2024,
doi: 10.23960/jitet.v12i1.3856.